IN THE CLAIMS

Please cancel claims 3, 4, 16, 17, 29, and 30.

Please amend claims 1, 2, 5, 8-12, 14, 15, 18, 21-25, 27, 28, 31, 34, 35, 37, 38, and 40 as follows.

1.   (Currently Amended) In a computer system, a method comprising:

generating obtaining one or more server actors on a server, wherein each actor is a functional component of a distributed application;

linking said server actors in a source first hierarchical tree;

generating a client actor tree representation dataset corresponding to a second hierarchical tree, wherein the second hierarchical tree is a subset of said server actors the first hierarchical tree;

sending said actor tree representation dataset to said a client; and

creating a member replicating the second hierarchical tree in said client using said actor tree representation dataset.

2.   (Currently Amended) The method of claim 1,

wherein each node in said source first hierarchical tree comprises a source actor;

wherein each node in the replicated second hierarchical tree comprises a member actor; and

wherein each member actor corresponds to a respective source actor.

3.   (Cancelled)

4.   (Cancelled)

5.   (Currently Amended) The method of claim 2[[4]], further comprising:

providing a communication interface between each member actor and its complementary corresponding source actor.

6.   (Original) The method of claim 1, wherein each said actor comprises a tree of hierarchically linked nodes, said nodes comprising one or more objects.

7.   (Original) The method of claim 6, wherein said nodes further comprise one or more nested actors.

8.   (Currently Amended) The method of claim 1, wherein said sending said ~~actor tree representation~~ dataset comprises sending said ~~tree representation~~ dataset via a secure communication network.

9.   (Currently Amended) The method of claim 2<u>3</u>, wherein said generating a ~~client actor tree representation~~ dataset comprises:

obtaining inclusion criteria from one or more parameter sets;

traversing said ~~source~~<u>first</u> hierarchical tree to determine nodes of said ~~source~~<u>first</u> hierarchical tree that comply with said inclusion criteria;

obtaining a pre-initialized object for each of said nodes that comply with said inclusion criteria;

generating a client graph comprising said pre-initialized objects.

10.   (Currently Amended) The method of claim 9, wherein said traversing said ~~source~~ <u>first</u> hierarchical tree is on a node-by-node basis starting from the root node and proceeding through all the leaf nodes.

11.   (Currently Amended) The method of claim 9<u>3</u>, wherein said ~~client actor tree representation comprises~~ <u>dataset is indicative of</u> the full client graph.

12.   (Currently Amended) The method of claim 2<u>3</u>, wherein said ~~client actor tree representation~~ dataset comprises a subgraph for updating ~~existing~~ <u>the replicated second hierarchical</u> ~~actor~~tree of said client.

13.   (Original) The method of claim 9, wherein said pre-initialized object comprises methods and attributes for construction and initialization of said client graph.

14.   (Currently Amended) A computer program product comprising:

a computer readable medium having computer program code embodied therein for creating and deploying client side actors for a server application, said computer readable medium comprising computer program code configured to cause a computer to:

generate ~~obtain~~ one or more ~~server~~ actors on a server, wherein each actor is a functional component of a distributed application;

link said ~~server~~ actors in a ~~source~~ first hierarchical tree;

generate a ~~client actor tree representation~~ dataset corresponding to a second hierarchical tree, wherein the second hierarchical tree is a subset of ~~said server actors~~ the first hierarchical tree;

send said ~~actor tree representation~~ dataset to ~~said~~ a client; and

~~create a member~~ replicate the second hierarchical tree in said client using said ~~actor tree representation~~ dataset.

15.   (Currently Amended) The computer program product of claim 14,

wherein each node in said ~~source~~ first hierarchical tree comprises a source actor;

wherein each node in the replicated second hierarchical tree comprises a member actor; and

wherein each member actor corresponds to a respective source actor.

16.   (Cancelled)

17.   (Cancelled)

18.   (Currently Amended) The computer program product of claim 15~~17~~, further comprising:

computer program code configured to provide a communication interface between each

member actor and its ~~complementary~~ corresponding source actor.

19.   (Original) The computer program product of claim 14, wherein each said actor comprises a tree of hierarchically linked nodes, said nodes comprising one or more objects.

20.   (Original) The computer program product of claim 19, wherein said nodes further comprise one or more nested actors.

21.   (Currently Amended) The computer program product of claim 14, wherein said send said ~~actor tree representation~~ dataset comprises sending said ~~tree representation~~ dataset using a secure communication network.

22.   (Currently Amended) The computer program product of claim 15~~16~~, wherein said generate a ~~client actor tree representation~~ dataset comprises:

    obtaining inclusion criteria from a parameter set;

    traversing said ~~source~~ first hierarchical tree to determine nodes of said ~~source~~ first
        hierarchical tree that comply with said inclusion criteria;

    obtaining a pre-initialized object for each of said nodes that comply with said inclusion
        criteria;

    generating a client graph comprising said pre-initialized objects.

23.   (Currently Amended) The computer program product of claim 22, wherein said traversing said ~~source~~ first hierarchical tree is on a node-by-node basis starting from the root node and proceeding through all the leaf nodes.

24.   (Currently Amended) The computer program product of claim 22 ~~17~~, wherein said ~~client actor tree representation~~ dataset is indicative of ~~comprises~~ the full client graph.

25.   (Currently Amended) The computer program product of claim 15~~17~~, wherein said ~~client actor tree representation~~ dataset comprises a subgraph for updating ~~existing~~ the replicated second

hierarchical ~~actor~~-tree of said client.

26.   (Original) The computer program product of claim 21, wherein said pre-initialized object comprises methods and attributes for construction and initialization of said client graph.

27.   (Currently Amended) An apparatus comprising:

    ~~one or more clients;~~

    a server comprising one or more server actors linked in a source hierarchical tree,

        wherein each actor is a functional component of a distributed application;

    one or more clients, each of said clients coupled to the server via a respective communication interface;

    wherein said server is configured to:

        generat~~ing~~e a dataset ~~client tree representation~~ for each of said one or more clients, wherein each dataset corresponds to a respective subset of the source hierarchical tree;

        ~~a communication interface between said server and said each of said one or more clients, said server~~ sending each dataset to the respective client ~~said client tree representation~~ via the respective ~~said~~ communication interface~~,~~;

    wherein each of said one or more clients ~~creating a member~~ is configured to replicate a respective subset of the source hierarchical tree based on said dataset ~~client tree representation~~.

28.   (Currently Amended) The apparatus of claim 27,

    wherein each node in said source hierarchical tree comprises a source actor;

    wherein each node in each replicated subset of the source hierarchical tree comprises a member actor; and

    wherein each member actor corresponds to a respective source actor.

29.   (Cancelled)

30.   (Cancelled)

31.   (Currently Amended) The apparatus of claim 28~~30~~, further comprising:

an application program interface between each member actor and its ~~complementary~~ <u>corresponding</u> source actor.

32.   (Original) The apparatus of claim 27, wherein each said actor comprises a tree of hierarchically linked nodes, said nodes comprising one or more objects.

33.   (Original) The apparatus of claim 32, wherein said nodes further comprise one or more nested actors.

34.   (Currently Amended) The apparatus of claim 27, wherein <u>each of</u> said communication interface<u>s</u> is secured.

35.   (Currently Amended) The apparatus of claim 28~~29~~, wherein said generating a <u>dataset</u> ~~client tree representation~~ comprises:

obtaining inclusion criteria from a parameter set;

traversing said source hierarchical tree to determine nodes of said source hierarchical
    tree that comply with said inclusion criteria;

obtaining a pre-initialized object for each of said nodes that comply with said inclusion
    criteria;

generating a client graph comprising said pre-initialized objects.

36.   (Original) The apparatus of claim 35, wherein said traversing said source hierarchical tree is on a node-by-node basis.

37.   (Currently Amended) The apparatus of claim <u>35</u> ~~29~~, wherein said ~~client actor~~

~~tree representation~~ dataset is indicative of ~~comprises~~ the full client graph.

38.   (Currently Amended) The apparatus of claim <u>28</u>~~29~~, wherein said ~~client actor tree~~

~~representation~~ dataset comprises a subgraph for updating ~~existing~~ <u>the replicated subset of</u>

<u>the source hierarchical</u> ~~actor~~ tree of said client.

39.   (Original) The apparatus of claim 35, wherein said pre-initialized object comprises

methods and attributes for construction and initialization of said client graph.

40.   (Currently Amended) An apparatus comprising:

      means for <u>generating</u> ~~obtaining~~ one or more ~~server~~ actors <u>on a server, wherein each actor</u>

          <u>is a functional component of a distributed application</u>;

      means for linking said ~~server~~ actors in a ~~source~~ <u>first</u> hierarchical tree;

      means for generating a ~~client actor tree representation~~ dataset corresponding to <u>a second</u>

          <u>hierarchical tree, wherein the second hierarchical tree is</u> a subset of ~~said server~~

          ~~actors~~ <u>the first hierarchical tree</u>;

      means for sending said ~~actor tree representation~~ dataset to ~~said~~ <u>a</u> client; and

      means for ~~creating a member~~ <u>replicating the second</u> hierarchical tree in said client

          using said ~~actor tree representation~~ dataset.